

IPNAT Updating

Share DNS news

Mon, Feb 1, 1999

Each IRM/local station currently maintains its IP Node Address Table by accessing the site-wide Domain Name Server. Each entry is updated once per 8 hours, approximately. But if we assume 100 nodes are doing this, with 100 entries each, this amounts to 20 accesses to the site-wide DNS every minute. This note describes how this effort could optionally be centralized to reduce this load on the DNS.

The DNSQ local application performs this updating service. It also accepts system-initiated requests for new nodes to be added to the IPNAT. The table grows over time, as no provision is made for deleting entries from the table, except manually. The maximum size of the table is about 250 entries. The table is in nonvolatile memory, so it isn't even cleared upon system reset or power-on. Each entry maintains its own set of time counters, so that it triggers an updating access to the DNS separate from any other. The time-outs are forced to be at least 1 second apart so as not to swamp the DNS with requests.

Solution 1

Add support in DNSQ for DNS requests. Then it would only be necessary to configure most nodes to access a central node for this information. The support would probably be limited to accepting requests for node names of the form `nodexxxx`, where `xxxx` is a node number in the IPNAT. If a request is received for a `node#` not in the central node's IPNAT, it would need to pass it to the real DNS, after which any answer would need to be passed back. This would be elegant, but it may also require a significant effort.

Solution 2

Design a new listype to support this kind of sharing of IPNAT entries. (One that currently exists accesses IPNAT entries via `entry#`, not `node#`.) The ident could specify a `node#`, and the data would be the IP address. Another approach would be a new listype whose setting permitted sending multiple pairs of `node#`/IP addresses. The centralized node might send out such a setting occasionally, or sooner if it received an update from the DNS that was a change in what it had in its IPNAT entry previously. But this would tend to synchronize the counters.

This solution should be easier to implement than the first, albeit less elegant. It is meant only to solve the congestion problem with DNS accesses, nothing more.